

Claims

- [c1] 1. A computer system having a tiered application comprising a business layer and an object-oriented data management layer that communicate via an application programming interface, and a database containing DICOM data that is managed by said data management layer, wherein an implementation of said business layer comprises code for requesting DICOM service via a synchronous message, and an implementation of said data management layer comprises code for providing DICOM service in response to said request, wherein said code in said data management layer employs asynchronous messaging to accomplish DICOM callback.
- [c2] 2. The computer system as recited in claim 1, wherein said data implementation layer is distributed on first and second computers connected via a network.
- [c3] 3. The computer system as recited in claim 1, wherein said database is based on a directory of DICOM files.
- [c4] 4. The computer system as recited in claim 1, wherein said data management layer implementation comprises an initialization method for constructing, at runtime, underlying objects of said data management layer implementation, said underlying objects being a function of the type of database to be managed.
- [c5] 5. The computer system as recited in claim 4, wherein said database comprises DICOM files.
- [c6] 6. The computer system as recited in claim 4, wherein said database comprises an archive.
- [c7] 7. The computer system as recited in claim 4, wherein said database comprises a network object.
- [c8] 8. The computer system as recited in claim 1, wherein said implementation of said data management layer comprises objects of a class having a method for counting the number of references to each object.
- [c9] 9. The computer system as recited in claim 8, wherein said objects have a

method for caching references.

- [c10] 10. The computer system as recited in claim 9, wherein implementation of said data management layer comprises objects of a session class having a method for clearing a cache in response to a message from said implementation of said business layer.
- [c11] 11. The computer system as recited in claim 9, wherein said objects have a method for removing children that have no references.
- [c12] 12. A method for writing a DICOM application, comprising the step of writing code for a business layer of a tiered application, said code sending synchronous messages that conform to a pre-existing application programming interface when DICOM service is requested by a user.
- [c13] 13. A workstation for viewing DICOM images, comprising a computer, a display monitor and an operator interface, wherein said computer is programmed with a tiered application comprising an object-oriented implementation of a data management layer that handles DICOM callbacks in response to requests for DICOM service having a format required by an application programming interface.
- [c14] 14. The workstation as recited in claim 13, wherein said implementation of said data management layer comprises code for sending asynchronous messages requesting DICOM service and said request is in the format of a synchronous message.
- [c15] 15. The workstation as recited in claim 13, wherein said implementation of said data management layer comprises objects of a class having a method for counting the number of references to each object.
- [c16] 16. The workstation as recited in claim 15, wherein said objects have a method for caching references.
- [c17] 17. The workstation as recited in claim 16, wherein said implementation of said data management layer comprises objects of a session class having a method for clearing a cache in response to an input to said operator interface.

[c18] 18. The workstation as recited in claim 16, wherein said objects have a method for removing children that have no references.

[c19] 19. A computer system comprising a database containing DICOM data and a tiered application, wherein said tiered application comprises:
a first tier comprising code for requesting a DICOM service by sending a synchronous message in a format conforming to an application programming interface; and
a second tier comprising code for managing said database to provide DICOM service in response to receipt of said request, wherein said code for managing said database sends asynchronous messages that implement a DICOM callback mechanism,
wherein said application programming interface represents a seam between said first tier code and said second tier code.

[c20] 20. A computer network comprising a service class user and a service class provider connected via a local area network, a database accessible by said service class provider, and a tiered application distributed between said service class user and said service class provider, wherein said tiered application comprises:
a first tier comprising code for requesting a DICOM service by sending a synchronous message in a format conforming to an application programming interface; and
a second tier comprising code for managing said database to provide DICOM service in response to receipt of said request, wherein said code for managing said database sends asynchronous messages that implement a DICOM callback mechanism,
wherein said application programming interface represents a seam between said first tier code and said second tier code.